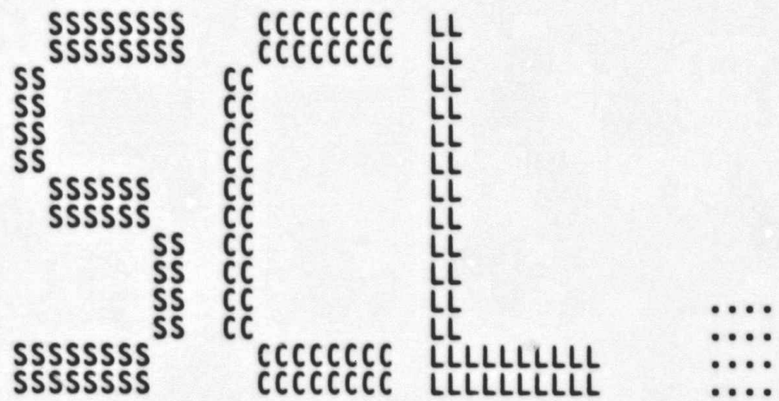
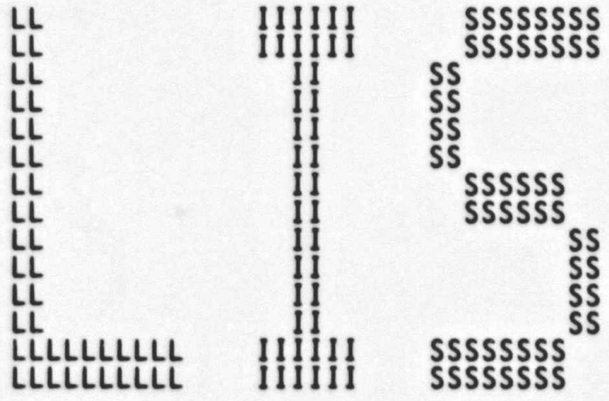
```
RRRRRRRRRRR      UUU          UUU  NNN          NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN          NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN          NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRR          RRR UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN       NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN       NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN       NNN  000         000  FFF              FFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN     NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRR    RRR       UUU          UUU  NNN       NNNNNN  000         000  FFF              FFF
RRR    RRR       UUU          UUU  NNN       NNNNNN  000         000  FFF              FFF
RRR    RRR       UUU          UUU  NNN       NNNNNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN          NNN  000         000  FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN          NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN          NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN          NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN          NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN          NNN     000000000     FFF              FFF
```

```
_$2

Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR
STR

STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

FILEID**SCL





SETD
V04-

1
1
1
1
1
1
1
1
1
1
1

```
;  1          0001  0 %TITLE 'Shift current word in MRA to the left'
;  2          0002  0 MODULE SCL (
;  3          0003  0                 IDENT = 'V04-000'
;  4        P 0004  0 %BLISS32[,
;  5        P 0005  0         ADDRESSING_MODE(EXTERNAL=LONG_RELATIVE,NONEXTERNAL=LONG_RELATIVE)
;  6          0006  0         ]
;  7          0007  0                 ) =
;  8          0008  1 BEGIN
;  9          0009  1 !
; 10          0010  1 !*****************************************************************************
; 11          0011  1 !*                                                                           *
; 12          0012  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
; 13          0013  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
; 14          0014  1 !*  ALL RIGHTS RESERVED.                                                     *
; 15          0015  1 !*                                                                           *
; 16          0016  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
; 17          0017  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
; 18          0018  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
; 19          0019  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
; 20          0020  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
; 21          0021  1 !*  TRANSFERRED.                                                             *
; 22          0022  1 !*                                                                           *
; 23          0023  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
; 24          0024  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
; 25          0025  1 !*  CORPORATION.                                                             *
; 26          0026  1 !*                                                                           *
; 27          0027  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
; 28          0028  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
; 29          0029  1 !*                                                                           *
; 30          0030  1 !*                                                                           *
; 31          0031  1 !*****************************************************************************
; 32          0032  1 !
; 33          0033  1 !++
; 34          0034  1 ! FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
; 35          0035  1 !
; 36          0036  1 ! ABSTRACT: Rearrange the MRA by moving the current word to the left.
; 37          0037  1 !
; 38          0038  1 !
; 39          0039  1 ! ENVIRONMENT: Transportable
; 40          0040  1 !
; 41          0041  1 ! AUTHOR: R.W.Friday     CREATION DATE: May, 1978
; 42          0042  1 !
```

```
;    44          0043  1 %SBTTL 'Revision History'
;    45          0044  1 !
;    46          0045  1 ! MODIFIED BY:
;    47          0046  1 !
;    48          0047  1 !       002     REM00002        Ray Marshall    07-Mar-1983
;    49          0048  1 !               Global edit of all modules. Updated module names, idents,
;    50          0049  1 !               copyright dates. Changed require files to BLISS library.
;    51          0050  1 !
;    52          0051  1 !--
```

```
;    54          0052  1 %SBTTL 'Module Level Declarations'
;    55          0053  1 !
;    56          0054  1 ! TABLE OF CONTENTS:
;    57          0055  1 !
;    58          0056  1 !
;    59          0057  1 ! INCLUDE FILES:
;    60          0058  1 !
;    61          0059  1
;    62          0060  1 LIBRARY 'NXPORT:XPORT';                 ! XPORT Library
;    63          0061  1 REQUIRE 'REQ:RNODEF';                   ! RUNOFF variant definitions
;    64          0192  1
;    65        U 0193  1 %IF DSRPLUS %THEN
;    66        U 0194  1 LIBRARY 'REQ:DPLLIB';                   ! DSRPLUS BLISS Library
;    67          0195  1 %ELSE
;    68          0196  1 LIBRARY 'REQ:DSRLIB';                   ! DSR BLISS Library
;    69          0197  1 %FI
;    70          0198  1
;    71          0199  1 !
;    72          0200  1 ! MACROS:
;    73          0201  1 !
;    74          0202  1 !
;    75          0203  1 ! EQUATED SYMBOLS:
;    76          0204  1 !
;    77          0205  1 !
;    78          0206  1 ! OWN STORAGE:
;    79          0207  1 !
;    80          0208  1 !
;    81          0209  1 ! EXTERNAL REFERENCES:
;    82          0210  1 !
;    83          0211  1
;    84          0212  1 EXTERNAL
;    85          0213  1     MRA : REF FIXED_STRING,
;    86          0214  1     SCA : SCA_DEFINITION,
;    87          0215  1     TSF : TSF_DEFINITION;
;    88          0216  1
;    89          0217  1 EXTERNAL ROUTINE
;    90          0218  1     GCSKIP;
;    91          0219  1
```

```
;    93     0220  1 %SBTTL 'SCL  --'
;    94     0221  1 GLOBAL ROUTINE scl : NOVALUE =
;    95     0222  1
;    96     0223  1 !++
;    97     0224  1 ! FUNCTIONAL DESCRIPTION:
;    98     0225  1 !
;    99     0226  1 ! See ABSTRACT for a sufficient explanation.
;   100     0227  1 !
;   101     0228  1 ! FORMAL PARAMETERS:     None
;   102     0229  1 !
;   103     0230  1 ! IMPLICIT INPUTS:       None
;   104     0231  1 !
;   105     0232  1 ! IMPLICIT OUTPUTS:      None
;   106     0233  1 !
;   107     0234  1 ! ROUTINE VALUE:
;   108     0235  1 ! COMPLETION CODES:      None
;   109     0236  1 !
;   110     0237  1 ! SIDE EFFECTS:          None
;   111     0238  1 !
;   112     0239  1 !--
;   113     0240  1
;   114     0241  2     BEGIN
;   115     0242  2
;   116     0243  2     LOCAL
;   117     0244  2         ptr,
;   118     0245  2         ptr_copy;
;   119     0246  2
;   120     0247  2     fs_init (mra);                          !reset pointers and counters back to start
;   121     0248  2     tsf_int_hl = 0;                         !Nothing in TSF yet.
;   122     0249  2     tsf_ext_hl = 0;                         !...
;   123     0250  2     tsf_int_vl = 0;                         !...
;   124     0251  2
;   125     0252  2     IF  .sca_wrd_int_l EQL 0  THEN          ! If no shift require, then
;   126     0253  2         RETURN;                             !   exit.
;   127     0254  2
;   128     0255  2     gcskip (.sca_spacing - 1);              !Generate intermediate code for skipping.
;   129     0256  2
;   130     0257  2     INCR i FROM 1 TO .sca_lm DO
;   131     0258  2         (fs_wchar (mra, %C' '));            !Fill out left margin with spaces.
;   132     0259  2
;   133     0260  2     tsf_int_hl = .tsf_int_hl + .sca_lm;
;   134     0261  2     tsf_ext_hl = .sca_lm;
;   135     0262  2     ptr = .fs_next (mra);                   !The current word will start here eventually.
;   136     0263  2     ptr_copy = .fs_next (mra);
;   137     0264  2
;   138     0265  2     INCR i FROM 1 TO (.sca_wrd_int_l) DO
;   139     0266  3         BEGIN                               !move the current word a character at a time.
;   140     0267  3         ! NOTE: This could have been coded using fs_wchar, and forgetting
;   141     0268  3         !       about using ptr_copy and hold_char; however, that uncovered a
;   142     0269  3         !       compiler bug.  Also, this generates more efficient object code.
;   143     0270  3
;   144     0271  3         LOCAL
;   145     0272  3             hold_char;
;   146     0273  3
;   147     0274  3         hold_char = CH$RCHAR_A (sca_wrd_pntr);
;   148     0275  3         CH$WCHAR_A (.hold_char, ptr_copy);
;   149     0276  3         fs_length (mra) = .fs_length (mra) + 1;
```

```
0277 2          END;
0278 2
0279 2      fs_next (mra) = .ptr_copy;
0280 2      sca_wrd_pntr = .ptr;                                         !re-establish pointer to SCA_WRD.
0281 2      RETURN;
0282 1      END;                                        !End of SCL

                                                        .TITLE  SCL Shift current word in MRA to the left
                                                        .IDENT  \V04-000\

                                                        .EXTRN  MRA, SCA, TSF, GCSKIP

                                                        .PSECT  $CODE$,NOWRT,2

                                  03FC 00000           .ENTRY  SCL, Save R2,R3,R4,R5,R6,R7,R8,R9          ; 0221
           59 00000000G   EF    9E 00002           MOVAB   TSF, R9                                    ;
           58 00000000G   EF    9E 00009           MOVAB   MRA, R8                                    ;
           57 00000000G   EF    9E 00010           MOVAB   SCA+248, R7                                ;
           50               68  D0 00017           MOVL    MRA, R0                                    ; 0247
                      0C   A0   D4 0001A           CLRL    12(R0)                                     ;
           60         10   A0   9E 0001D           MOVAB   16(R0), (R0)                               ;
      04   A0              60   D0 00021           MOVL    (R0), 4(R0)                                ;
           50              69   D0 00025           MOVL    TSF, R0                                    ;
                           60   7C 00028           CLRQ    (R0)                                       ; 0248
                      18   A0   D4 0002A           CLRL    24(R0)                                     ; 0250
                      04   A7   D5 0002D           TSTL    SCA+252                                    ; 0252
                           5A   13 00030           BEQL    5$                                         ;
 7E        84   B7         01   C3 00032           SUBL3   #1, @SCA+124, -(SP)                        ; 0255
   00000000G    EF         01   FB 00037           CALLS   #1, GCSKIP                                 ;
           53       FF7C   D7   D0 0003E           MOVL    @SCA+116, R3                               ; 0257
           50              68   D0 00043           MOVL    MRA, R0                                    ; 0258
           52         04   A0   9E 00046           MOVAB   4(R0), R2                                  ;
           51              68   D0 0004A           MOVL    MRA, R1                                    ;
                           50   D4 0004D           CLRL    I                                          ;
                           09   11 0004F           BRB     2$                                         ;
      00   B2              20   90 00051 1$:       MOVB    #32, @0(R2)                                ;
                           62   D6 00055           INCL    (R2)                                       ;
                      0C   A1   D6 00057           INCL    12(R1)                                     ;
 F3        50              53   F3 0005A 2$:       AOBLEQ  R3, I, 1$                                  ; 0257
           50              69   D0 0005E           MOVL    TSF, R0                                    ; 0258
           60              53   C0 00061           ADDL2   R3, (R0)                                   ; 0260
      04   A0              53   D0 00064           MOVL    R3, 4(R0)                                  ; 0261
           56              62   D0 00068           MOVL    (R2), PTR                                  ; 0262
           50              62   D0 0006B           MOVL    (R2), PTR_COPY                             ; 0263
           54         04   A7   D0 0006E           MOVL    SCA+252, R4                                ; 0265
                           53   D4 00072           CLRL    I                                          ;
                           0C   11 00074           BRB     4$                                         ;
           55         00   B7   9A 00076 3$:       MOVZBL  @SCA+248, HOLD_CHAR                        ; 0274
                           67   D6 0007A           INCL    SCA+248                                    ;
           80              55   90 0007C           MOVB    HOLD_CHAR, (PTR_COPY)+                     ; 0275
                      0C   A1   D6 0007F           INCL    12(R1)                                     ; 0276
 F0        53              54   F3 00082 4$:       AOBLEQ  R4, I, 3$                                  ; 0265
           62              50   D0 00086           MOVL    PTR_COPY, (R2)                             ; 0279
           67              56   D0 00089           MOVL    PTR, SCA+248                               ; 0280
                                04 0008C 5$:       RET                                                ; 0282
```

```
; Routine Size:  141 bytes,    Routine Base:  $CODE$ + 0000

;  156          0283  1
;  157          0284  1 END                                              !End of module
;  158          0285  0 ELUDOM



;                                PSECT SUMMARY
;
;       Name                      Bytes                      Attributes
;
;   $CODE$                            141  NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)



;                           Library Statistics
;
;                                               -------- Symbols --------       Pages      Processing
;       File                                    Total    Loaded   Percent      Mapped     Time
;
;  _$255$DUA28:[SYSLIB]XPORT.L32;1                590         0         0         252       00:00.1
;  _$255$DUA28:[RUNOFF.SRC]DSRLIB.L32;1          1248        20         1          86       00:00.2



;                                COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:SCL/OBJ=OBJ$:SCL MSRC$:SCL/UPDATE=(ENH$:SCL)

; Size:         141 code + 0 data bytes
; Run Time:        00:04.9
; Elapsed Time:    00:17.6
; Lines/CPU Min:     3482
; Lexemes/CPU-Min: 16460
; Memory Used:  57 pages
; Compilation Complete
```